

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-006170
(43)Date of publication of application : 10.01.2003

(51)Int.Cl. G06F 15/16
G06F 9/445

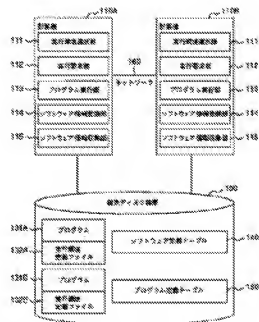
(21)Application number : 2001-185848 (71)Applicant : HITACHI LTD
(22)Date of filing : 20.06.2001 (72)Inventor : NITTA KUNIHISA

(54) METHOD FOR PERFORMING PROGRAM IN ENVIRONMENT OF PLURAL COMPUTERS

(57)Abstract:

PROBLEM TO BE SOLVED: To perform a program without making a user be aware of a computer, when a program depending on any computer is executed, in a system composed of a plurality of computers.
SOLUTION: A performance environment definition file 132 stores software information constituting a performance environment in correspondence with the program 131. A software definition table 140 stores the information of software constituting a performance environment according to the respective computers. A performance environment selecting part 111 refers to the performance environment definition file 132, corresponding to the program 131 to be performance object to acquire performance environment information and selects a computer 110, having performance environment information which includes the acquired performance environment information by referring to the software definition table 140. A performance requesting part 112 demands the selected computer 110 to execute the program, and a program executing part 113 of an object computer execute the program.

図 1



(51) Int.Cl. ⁷	識別記号	F I	データベース* (参考)
G 0 6 F 15/16	6 2 0	G 0 6 F 15/16	6 2 0 A 5 B 0 4 5
9/445		9/06	6 5 0 C 5 B 0 7 6

審査請求 未請求 請求項の数10 O L (全 7 頁)

(21) 出願番号 特願2001-185848(P2001-185848)

(22) 出願日 平成13年6月20日 (2001.6.20)

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 新田 邦久

神奈川県横浜市戸塚区戸塚町5030番地 株
式会社日立製作所ソフトウェア事業部内

(74) 代理人 100068504

弁理士 小川 勝男 (外2名)

Fターム(参考) 5B045 G002 JJ08

5B076 A817

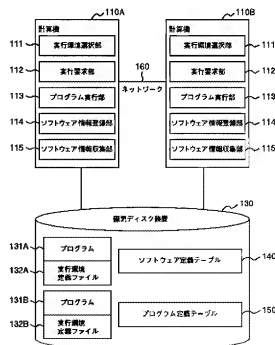
(54) 【発明の名称】 複数計算機環境でのプログラム実行方法

(57) 【要約】

【課題】 複数の計算機で構成されるシステムにおいて、いずれかの計算機に依存するプログラムを実行する際に、ユーザに計算機を意識させることなくプログラムを実行する。

【解決手段】 実行環境定義ファイル132は、プログラム131に対応して実行環境を構成するソフトウェア情報を格納する。ソフトウェア定義テーブル140は、計算機の各々に対応して実行環境を構成するソフトウェアの情報を格納する。実行環境選択部111は、実行対象とするプログラム131に対応する実行環境定義ファイル132を参照して実行環境情報を取得し、ソフトウェア定義テーブル140を参照して取得した実行環境情報を包含する実行環境情報を有する計算機110を選択する。実行要求部112は、選択された計算機110にそのプログラムの実行を要求し、対象計算機のプログラム実行部113がそのプログラムを実行する。

図 1



【特許請求の範囲】

【請求項1】複数の計算機で構成され前記計算機のいずれか少なくとも1つの計算機に依存するプログラムを有する複数の計算機環境でのプログラム実行方法であって、実行対象とするプログラムに対応して実行環境を構成するソフトウェアの情報を格納する第1の記憶手段を参照して該当するソフトウェア情報を取得し、前記計算機の各々に対応して実行環境を構成するソフトウェアの情報を格納する第2の記憶手段を参照して取得された前記ソフトウェア情報を包含するソフトウェアの情報を有する計算機を選択し、選択された前記計算機に実行対象とする前記プログラムの実行を要求し、選択された前記計算機によって前記プログラムを実行することを特徴とする複数の計算機環境でのプログラム実行方法。

【請求項2】前記計算機は論理計算機であることを特徴とする請求項1記載の複数の計算機環境でのプログラムの実行方法。

【請求項3】実行対象となった前記プログラムの識別子と選択された前記計算機の識別子との対応を第3の記憶手段に登録し、前記ソフトウェア情報を取得する前に前記第3の記憶手段を参照してすでに対象とするプログラムに対応する計算機の識別子が登録されていれば登録済の計算機に対象とする前記プログラムの実行を要求することを特徴とする請求項1記載の複数の計算機環境でのプログラムの実行方法。

【請求項4】前記第1の記憶手段に格納される対応情報は、前記プログラムの名称と前記ソフトウェア情報を格納するファイルの名称について名称構成によって関連付けられることを特徴とする請求項1記載の複数の計算機環境でのプログラムの実行方法。

【請求項5】複数の計算機で構成され前記計算機のいずれか少なくとも1つの計算機に依存するプログラムを有する複数の計算機環境でのプログラム実行方法であって、実行対象とするプログラムの識別子と実行対象とする前記プログラムを実行可能な前記計算機の識別子との対応についての情報を格納する記憶手段を参照してすでに対象とする前記プログラムに対応する計算機の識別子が登録されていれば登録済の計算機に対象とする前記プログラムの実行を要求し、登録されていなければ前記計算機のいずれか1つに実行対象とする前記プログラムの実行を要求し、実行要求した前記計算機から実行成功の報告があったとき、実行対象となった前記プログラムの識別子と実行成功を報告した前記計算機の識別子との対応についての情報を前記記憶手段に登録し、実行要求した前記計算機から実行不成功の報告があったとき、次の前記計算機について実行対象とする前記プログラムの実行を要求することを特徴とする複数の計算機環境でのプログラム実行方法。

【請求項6】複数の計算機で構成され前記計算機のいずれか少なくとも1つの計算機に依存するプログラムを有

する複数の計算機環境において実行対象のプログラムの実行制御をするプログラムであって、コンピュータに、実行対象とするプログラムに対応して実行環境を構成するソフトウェアの情報を格納する第1の記憶手段を参照して該当するソフトウェア情報を取得する手順、前記計算機の各々に対応して実行環境を構成するソフトウェアの情報を格納する第2の記憶手段を参照して取得された前記ソフトウェア情報を包含するソフトウェアの情報を有する計算機を選択する手順、および選択された前記計算機に実行対象とする前記プログラムの実行を要求する手順を実行させるための実行対象のプログラムの実行制御をするプログラム。

【請求項7】前記計算機は論理計算機であることを特徴とする請求項6記載のプログラム。

【請求項8】さらに前記コンピュータに、実行対象となった前記プログラムの識別子と選択された前記計算機の識別子との対応を第3の記憶手段に登録する手順と、前記ソフトウェア情報を取得する前に前記第3の記憶手段を参照してすでに対象とするプログラムに対応する計算機の識別子が登録されていれば登録済の計算機に対象とする前記プログラムの実行を要求する手順とを実行させることを特徴とする請求項6記載のプログラム。

【請求項9】前記第1の記憶手段に格納される対応情報は、前記プログラムの名称と前記ソフトウェア情報を格納するファイルの名称について名称構成によって関連付けられることを特徴とする請求項6記載のプログラム。

【請求項10】複数の計算機で構成され前記計算機のいずれか少なくとも1つの計算機に依存するプログラムを有する複数の計算機環境において実行対象のプログラムの実行制御をするプログラムであって、コンピュータに、実行対象とするプログラムの識別子と実行対象とする前記プログラムを実行可能な前記計算機の識別子との対応についての情報を格納する記憶手段を参照してすでに対象とする前記プログラムに対応する計算機の識別子が登録されていれば登録済の計算機に対象とする前記プログラムの実行を要求する手順、登録されていなければ前記計算機のいずれか1つに実行対象とする前記プログラムの実行を要求する手順、実行要求した前記計算機から実行成功の報告があったとき、実行対象となった前記プログラムの識別子と実行成功を報告した前記計算機の識別子との対応についての情報を前記記憶手段に登録する手順、および実行要求した前記計算機から実行不成功の報告があったとき、次の計算機について実行対象とする前記プログラムの実行を要求する手順を実行させるための実行対象のプログラムの実行制御をするプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、複数の計算機で構成され、いずれかの計算機の環境に依存するプログラムを有する複数の計算機環境でのプログラム実行方法に関す

る。

【0002】

【従来の技術】複数の計算機をネットワーク等で接続し、処理機能を複数計算機に分散させ、複数計算機の連携によって処理を進める分散システムが知られている。そのようなシステムの1つであるクライアント・サーバ・システムは、クライアント側とサーバ側の各々のプログラム間でメッセージをやりとりすることにより、処理を進める。例えば「第2版TCP/IPによるネットワーク構築VOL.1ー原理・プロトコル・アーキテクチャ」P241～P247、Douglas Comer著 村井純・楠本博之訳 発行所 共立出版株式会社などがある。

【0003】また他の方法として、複数の計算機の中から実行する計算機を自動的に選択する方法がある。実行計算機を選択方法として、例えば特開平10-334057号公報「分散システム環境におけるパッチジョブの動的負荷分散処理方法およびそのシステム」がある。この方法は、各計算機のCPU使用率、メモリ使用率、I/O使用率などの資源使用特性から実行計算機を選択する。

【0004】

【発明が解決しようとする課題】しかし上記のクライアント・サーバ・モデルでは、クライアント側とサーバ側の両方にプログラムを設ける必要があるため、クライアントはどのサーバ上に目的のプログラムがあるかを意識する必要がある、プログラムの開発とその管理が煩雑になる。

【0005】また負荷分散処理方法は、ジョブが使用する資源の使用状況によって計算機を選択するものであるため、アーキテクチャやオペレーティングシステム、利用可能なソフトウェアが異なる複数の計算機から構成されるシステムで、実行対象のプログラムがいずれかの計算機の実行環境に依存する場合にはこの方法を適用することはできない。このように計算機に依存するプログラムは、そのプログラムが必要とする実行環境を備えている計算機を選択する必要がある。

【0006】本発明の目的は、このような計算機に依存するプログラムを実行する際に、ユーザにプログラムを実行する計算機を意識させることなく、プログラムを実行する方法を提供することにある。

【0007】

【課題を解決するための手段】本発明は、複数の計算機で構成されそれぞれ計算機のいずれか少なくとも1つの計算機に依存するプログラムを有する複数計算機環境でのプログラム実行方法であって、実行対象とするプログラムに対応して実行環境を構成するソフトウェアの情報を格納する第1の記憶手段を参照して該当するソフトウェア情報を取得し、計算機の各々に対応して実行環境を構成するソフトウェアの情報を格納する第2の記憶手段を

参照して取得されたソフトウェア情報を包含するソフトウェアの情報を有する計算機を選択し、選択された計算機に実行対象とするプログラムの実行を要求し、選択された計算機によって対象のプログラムを実行する複数計算機環境でのプログラム実行方法を特徴とする。

【0008】また本発明は、実行対象となったプログラムの識別子と選択された計算機の識別子との対応を第3の記憶手段に登録し、上記のソフトウェア情報を取得する前に第3の記憶手段を参照してすでに対象とするプログラムに対応する計算機の識別子が登録されている登録済みの計算機に対象とするプログラムの実行を要求する複数計算機環境でのプログラムの実行方法を特徴とする。

【0009】本発明により、ユーザは複数の計算機で構成される計算機システムを単一のシステムと同じように扱い、プログラムの開発と管理を行うことが可能となる。

【0010】

【発明の実施の形態】以下、本発明の実施形態について図面を用いて説明する。

【0011】図1は、本実施形態のシステム構成を示す図である。システムは、計算機110A、計算機110B、両計算機を接続するネットワーク160及び両計算機に接続され共用される磁気ディスク装置130から構成される。計算機110A及び計算機110Bは、各々そのメモリ上に実行環境選択部111、実行要求部112、プログラム実行部113、ソフトウェア情報登録部114及びソフトウェア情報収集部115の各プログラムを格納し、その処理装置によって実行する。

【0012】磁気ディスク装置130は、計算機110Aで作成したプログラム131Aと計算機110Bで作成したプログラム131B、プログラム131Aに対応する実行環境定義ファイル132A、プログラム131Bに対応する実行環境定義ファイル132B、ソフトウェア定義テーブル140及びプログラム定義テーブル150を格納しており、計算機110Aと計算機110Bから共有される。以下計算機110A又は計算機110Bのいずれかを指すときには計算機110のように呼ぶことがある。プログラム131及び実行環境定義ファイル132についても同様である。

【0013】図2は、ソフトウェア定義テーブル140に格納されるソフトウェア情報の例を示す図である。この例によれば、計算機110AのアーキテクチャはARCH-A、オペレーティングシステムはOS-A、使用可能なソフトウェアはSOFT-A1、SOFT-A2などであることを定義している。また計算機110BのアーキテクチャはARCH-B、オペレーティングシステムはOS-B、使用可能なソフトウェアはSOFT-B1、SOFT-B2などであることを定義している。ここでソフトウェアをソフトウェア名で示しており、そ

5

の実体は少なくとも1つのプログラムから成る。ソフトウェア定義テーブル140は、各計算機についてその計算機全体としてのプログラムの実行環境についての情報を提供する。

【0014】図3は、プログラム定義テーブル150のデータ構成を示す図である。プログラム定義テーブル150の各エントリは、プログラム名(識別子)とそのプログラムが実行可能な計算機の計算機名(識別子)との対応関係を示す。例えばプログラム131Bは、計算機110Bの実行環境の下で実行可能であることを示して

いる。

【0015】図4は、一般的な階層ファイルシステムにおいて、プログラムと実行環境ファイルを関連付ける例を説明する図である。ルート(/)71の下にディレクトリTMP72が存在し、その配下にプログラムEXEA73と実行環境定義ファイルEXEA.SPEC74とディレクトリTMPX75が存在する。TMPX75には、実行環境定義ファイルTMPX.SPEC76、プログラムEXEB77とプログラムEXEC78が存在する。EXEA73、EXEB77及びEXEC78は、バイナリ形式またはスクリプト形式の実行可能ファイルである。本例ではプログラムと実行環境定義ファイルをその名称によって関連付けており、実行環境定義ファイルは関連するプログラム名またはディレクトリ名の後に、SPECという語を付加して区別している。すなわちEXEA.SPEC74はプログラムEXEA73の実行環境定義ファイルであり、TMPX.SPEC76はディレクトリTMPX75の実行環境定義ファイルである。実行環境定義ファイルは、テキスト形式のファイルである。ディレクトリに対応する実行環境定義ファイルは、対応するディレクトリ内のファイル全てを関連付けるものである。すなわちTMPX75配下のプログラムEXEB77とEXEC78は同じ実行環境で実行可能であり、その実行環境は実行環境定義ファイルTMPX.SPEC76に定義されている。このような方法により実行環境定義ファイルを少ない作業量で設定することが可能である。いずれの計算機110の実行環境選択部111もこのようなディレクトリ及びファイルにアクセス可能である必要がある。

【0016】図5は、実行環境定義ファイル132Aおよび132Bの情報例を示す図である。実行環境定義ファイル132Aは、プログラム131Aを実行するためには、オペレーティングシステムO-S、ソフトウェアSOFT-A1、SOFT-A2が必要であることを定義している。これは計算機110Aのソフトウェア情報のサブセットになっている。またプログラム131Aは、アーキテクチャに依存しない。同様に実行環境定義ファイル132Bは、プログラム131Bを実行するために、アーキテクチャARCH-B、オペレーティングシステムO-S-B、ソフトウェアSOFT-B1が必要

6

であることを定義している。これは計算機110Bのソフトウェア情報のサブセットになっている。

【0017】各計算機のソフトウェア情報収集部115は、その計算機についてのソフトウェア情報を収集する。アーキテクチャやオペレーティングシステムは、オペレーティングシステムが持つ情報テーブルより取得可能であり、利用可能なソフトウェアはインストール情報ファイルから取得可能である。インストール情報ファイルとは、インストールされているソフトウェアの情報を管理するファイルであり、インストールに使用するインストール用プログラム(インストーラ)が作成し、更新するファイルである。各ソフトウェア情報登録部114は、その計算機が初めて磁気ディスク装置130と接続する際に、収集したソフトウェア情報をソフトウェア定義テーブル140に登録する。

【0018】図6は、プログラム実行時の実行環境選択部111及び実行要求部112の処理の流れを示すフローチャートである。プログラム131を実行する際に、実行環境選択部111は、プログラム定義テーブル150を検索して実行要求されたプログラムがこのテーブルに登録されているか判定する(ステップ500)。以前にプログラムが実行されている場合は、プログラム定義テーブル150にそのプログラム名が登録されているので、その登録されている計算機を選択する(ステップ501)。

【0019】目的のプログラムがプログラム定義テーブル150に登録されていない場合には、そのプログラム131に関連付けられた実行環境定義ファイル132を取得し(ステップ502)、ソフトウェア定義テーブル140を参照し、この実行環境定義ファイルの条件を満たす(実行環境を包含する)計算機を検索する(ステップ503)。条件を満たす計算機が存在しない場合は実行不可と判断し(ステップ505)、処理を終了する。存在する場合は条件を満たす計算機を選択する(ステップ504)。このような条件を満たす計算機が複数ある場合には、任意の(例えば最初の)計算機を選択する。実行要求部112は、選択した計算機に実行を要求し(ステップ506)、要求した計算機からの実行結果通知を待つ(ステップ507)。実行環境選択部111は、実行要求部112からプログラムの実行結果を受け取ることが可能である。実行環境選択部111は、実行結果通知を受け取った後、実行結果を判定する(ステップ508)。実行結果が成功の場合、ステップ500で目的のプログラムがプログラム定義テーブル150に登録されていないと判断しているならば(ステップ509未登録)、当該プログラムのプログラム名と計算機名との対応をプログラム定義テーブル150に登録し(ステップ510)、処理を終了する。ステップ508で実行結果が失敗か実行不可と判定した場合、及びステップ509で既に登録済みと判定された場合は処理を終了す

る。

【0020】図1の計算機110Aが計算機110B用に作成されたプログラム131Bを実行する場合を例として説明する。なおプログラム定義テーブル150には何も登録されておらず、ソフトウェア定義テーブル140は図2に示す状態であるとする。まずプログラム定義テーブル150にプログラム131Bが登録されているか判定する。登録されていないので、実行環境定義ファイル132Bを読み込み、ソフトウェア定義テーブル140より、アーキテクチャARCH-B、オペレーティングシステムOS-BとソフトウェアSOF-T-B1を備える計算機110Bを発見する。そして実行要求部112がネットワーク160経由で計算機110Bのプログラム実行部113に実行を要求する。計算機110Bのプログラム実行部113は、要求されたプログラムを実行し、その実行結果を計算機110Aに通知する。計算機110Aの実行要求部112は実行結果を受け取り、その実行結果を実行環境選択部111に渡す。実行結果が成功の場合はプログラム定義テーブル150に登録し、失敗の場合は登録しない。計算機110Aがプログラム131Bを計算機110Bに実行要求し、実行結果が成功の場合、プログラム定義テーブル150は図3のように登録される。なおプログラム定義テーブル150が存在すれば、ステップ502とステップ503の処理をバイパスできる場合があるが、プログラム定義テーブル150がなくても本発明を実施できる。

【0021】図7は、プログラム実行部113の処理の流れをなすフローチャートである。まずいずれかの計算機からの実行要求を待つ(ステップ801)。実行要求を受け取った場合には、要求されたプログラムが実行可能か判定する(ステップ802)。その判定内容は、プログラムへのアクセスは可能か、また実行権限を保持するか等、実際に実行する計算機から実行するプログラムへのアクセスに関するものである。ステップ802での判定の結果、プログラムを実行不可能な場合は、実行結果として実行不可を設定する(ステップ805)。判定結果が実行可能な場合は、要求されたプログラムを実行する(ステップ803)。その実行結果が成功の場合は(ステップ804成功)、実行結果に成功を設定し(ステップ806)、失敗の場合は実行結果に失敗を設定する(ステップ807)。その後、実行を要求した計算機に対し、実行結果を通知する(ステップ808)。

【0022】以上説明した処理手順により、計算機110Aと計算機110B間でユーザが計算機を意識することなく、計算機110Aから計算機110Bに依存するプログラム131Bを直接実行することが可能となる。逆に計算機110Bから計算機110Aのプログラムを実行することも可能である。

【0023】上記例では、実行環境定義ファイル132が登録されている場合について説明したが、実行環境定

義ファイル132が登録されていない場合の実行環境選択部111及び実行要求部112の処理手順について図8を用いて説明する。ステップ500でプログラム定義テーブル150に未登録の場合、ステップ502及び503の代わりにソフトウェア定義テーブル140の先頭に登録されている計算機を選択し(ステップ901)、その計算機にプログラムの実行を要求し(ステップ902)、その後実行結果の通知を待つ(ステップ903)。実行結果通知後、その実行結果を判定する(ステップ904)。実行結果が成功ならば(ステップ904成功)、プログラム名と計算機名との対応をプログラム定義テーブル150に登録し(ステップ905)、処理を終了する。実行結果が失敗か実行不可の場合、ソフトウェア定義テーブル140に登録されている計算機が他にあるか判定する(ステップ906)。計算機がある場合は、現存選択している計算機の次にソフトウェア定義テーブル140に登録されている計算機を選択し(ステップ907)、ステップ902に戻り再度プログラム実行を要求する。ソフトウェア定義テーブル140に次の計算機がない場合は、処理を終了する。例えば図1の計算機110Aが図2のソフトウェア定義テーブル140の状態では、実行環境定義ファイルが登録されていないプログラムを実行する場合、まず計算機110Aで実行を試みる。成功ならば、プログラム定義テーブル150に上記の対応情報を登録する。失敗した場合、計算機110Bで実行を試行し、成功ならばプログラム定義テーブル150に対応情報を登録する。そして次回以降は、プログラム定義テーブル150を参照して実行に成功した計算機を選択する。

【0024】以上説明した方法により、複数計算機間で実行する計算機を意識せずにプログラムを実行することが可能である。なお本実施形態では、計算機110Aと計算機110Bがネットワークで接続される場合について説明したが、ディスクシェア、メモリアレンジなどの手段によっても計算機間通信を行ってもよい。また計算機110Aおよび計算機110Bが各々論理計算機であっても本発明を支障なく実施できる。その場合には、ネットワーク160の代わりに論理計算機間の通信を制御する制御プログラムが必要となる。またプログラムと実行環境定義ファイルの関連付けをファイル名称によって行っているが、他の方法により関連付けを行っても良い。

【0025】

【発明の効果】以上説明したように本発明によれば、複数の計算機で構成された計算機システムにおいて、いずれかの計算機に依存するプログラムを有する複数計算機環境の中でプログラムを実行する際に、ユーザにプログラムを実行する計算機を意識させることのないプログラム実行制御が可能となる。また本発明によれば、プログラムの実行環境に適合する実行環境をもつ計算機を自動的に選択してその計算機によってプログラムを実行させ

るため、ユーザは単一の計算機と同じようにプログラムを作成、管理することが可能となる。

【図面の簡単な説明】

【図1】実施形態の複数の計算機より構成される計算機システムの構成図である。

【図2】実施形態のソフトウェア定義テーブル140の内容を示す図である。

【図3】実施形態のプログラム定義テーブル150の内容を示す図である。

【図4】プログラムと実行環境定義ファイルのファイルシステムを関連付ける例を示す図である。

【図5】実施形態の実行環境定義ファイル132の内容を示す図である。

【図6】実施形態においてプログラム実行時の実行環境選択部111及び実行要求部112の動作を説明するた

めのフローチャートである。

【図7】実施形態のプログラム実行部113の動作を説明するためのフローチャートである。

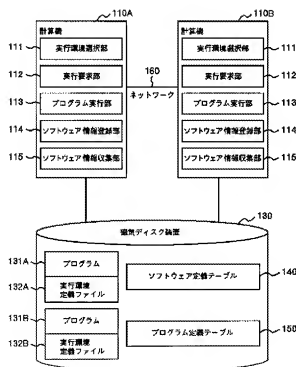
【図8】実施形態において実行環境定義ファイルが定義されていない場合の実行環境選択部111及び実行要求部112の動作を説明するフローチャートである。

【符号の説明】

110A、110B：計算機、111：実行環境選択部、112：実行要求部、113：プログラム実行部、114：ソフトウェア情報登録部、115：ソフトウェア情報収集部、131A、131B：プログラム、132A、132B：実行環境定義ファイル、140：ソフトウェア定義テーブル、150：プログラム定義テーブル

【図1】

図 1



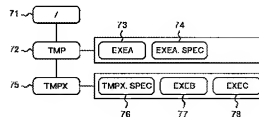
【図2】

図 2

計算機名	ソフトウェア情報
計算機 110A	ARCH: ARCH-A SOFT: SOFT-A3 OS: OS-A SOFT: SOFT-A4 SOFT: SOFT-A1 ! SOFT: SOFT-A2
計算機 110B	ARCH: ARCH-B SOFT: SOFT-B2 OS: OS-B SOFT: SOFT-B3 SOFT: SOFT-B1 !

【図4】

図 4



【図5】

図 5

【図3】

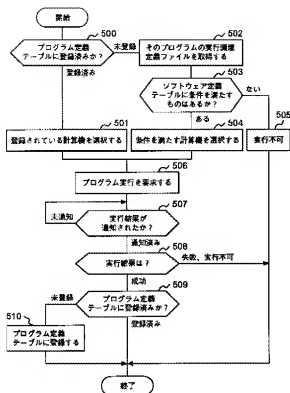
図 3

プログラム名	計算機名
プログラム 131B	計算機 110B
...	...

OS: OS-A SOFT: SOFT-A1 SOFT: SOFT-A2	132A
ARCH: ARCH-B OS: OS-B SOFT: SOFT-B1	132B

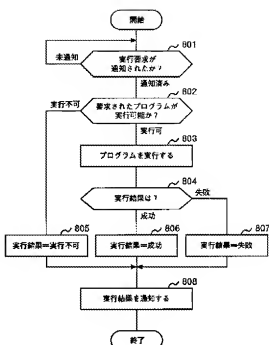
【図6】

図 6



【図7】

図 7



【図8】

図 8

